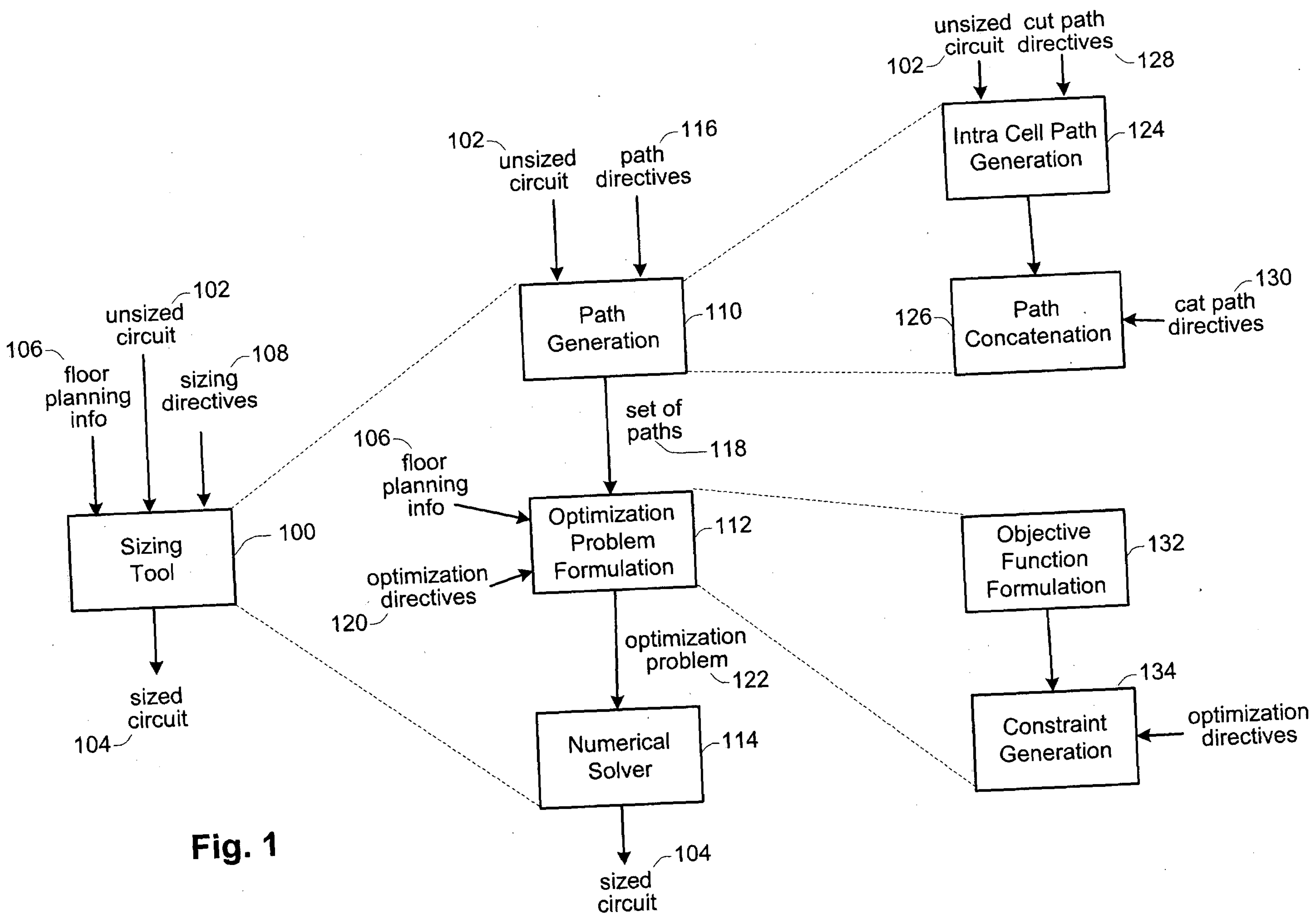
unsized circuit
cut path directives
102
128
Intra Cell Path Generation
124
102
unsized circuit
path directives
116
Path Generation
110
Path Concatenation
126
cat path directives
130
102
unsized circuit
106
floor planning info
sizing directives
108
set of paths
118
106
floor planning info
Optimization Problem Formulation
112
Objective Function Formulation
132
Sizing Tool
100
optimization directives
120
optimization problem
122
134
Constraint Generation
optimization directives
sized circuit
104
Numerical Solver
114
104
sized circuit

Fig. 1

200

```
        SizingPathGeneration(set1)
            foreach leaf cell cella in set1
                find all the half-operators in cella  that are going to start a path, put in seta
                foreach half-operator hoa in seta
                    lsta is an empty list of half-operators
                    RecursivePaths(lsta, hoa)

        RecursivePaths(lst1, ho1)
            If (lst1 contains ho1) make sizing-path from lst1, then return        // found cycle
            If (ho1 drives an observable point) make sizing-path from lst1 and ho1, then return
            foreach hoa that is driven by ho1 and has opposite driving direction of ho1
RecursivePaths(lst1+ho1, hoa)
```

Fig. 2

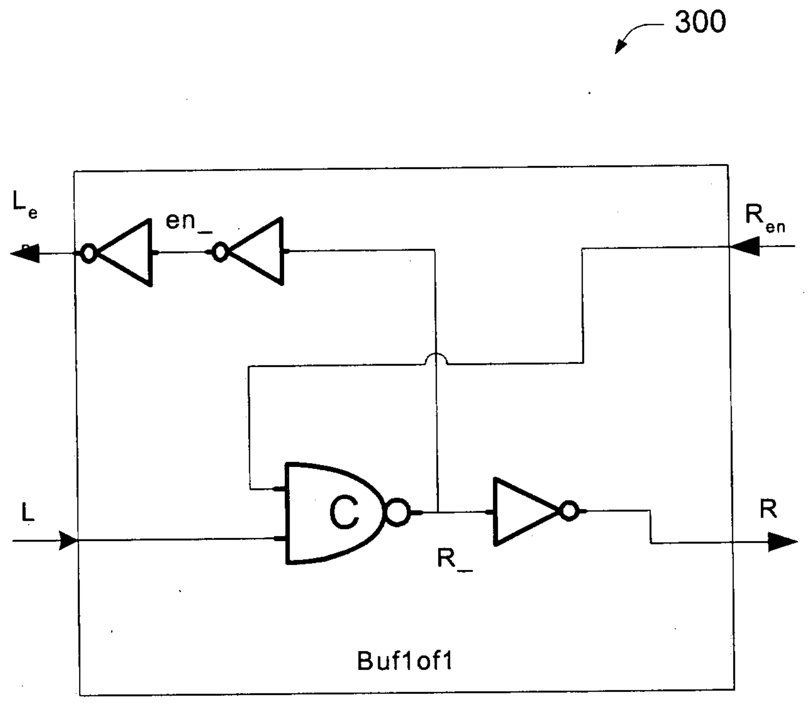

Fig. 3

400

```
CatPathGeneration(set1)
    lsta is sorted list of mid-level cells in set1, from low to high (e.g. from mid  to top levels)
    foreach cella in lsta
        // N.B. an nonobservable sucells is a subcell which contains
        // a nonobservable point
        pop up paths from nonobservable subcells of cella to cella
        find all the paths in cella that are going to start a cat-path, put in seta
        foreach patha in seta
            lsta is an empty list of paths
            RecursiveCatPaths(lsta, patha)

RecursiveCatPaths(lst1, path1)
    if (lst1 contains path1) make cat-path from lst1, then return      // found cycle
    if (last half-operator of path1 drives  an observable point)
        make cat-path from lst1 and path1, then return
    foreach patha that is driven by path1
RecursiveCatPaths(lst1+path1, patha)
```

Fig. 4

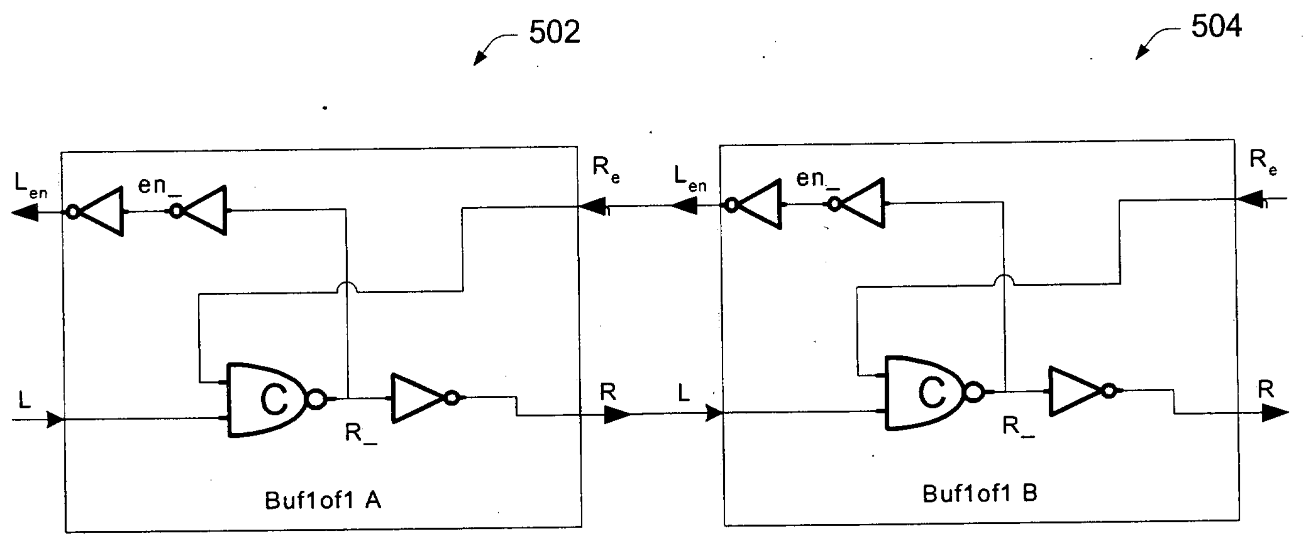

Fig. 5

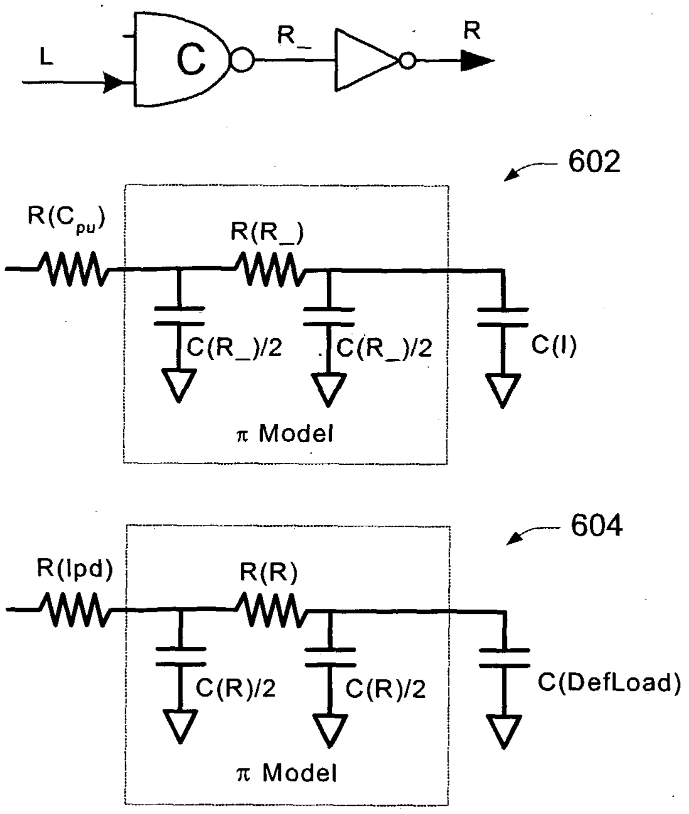
L
C
R_
R
602
R(C$_{pu}$)
R(R_)
C(R_)/2
C(R_)/2
C(l)
π Model
604
R(Ipd)
R(R)
C(R)/2
C(R)/2
C(DefLoad)
π Model

Fig. 6

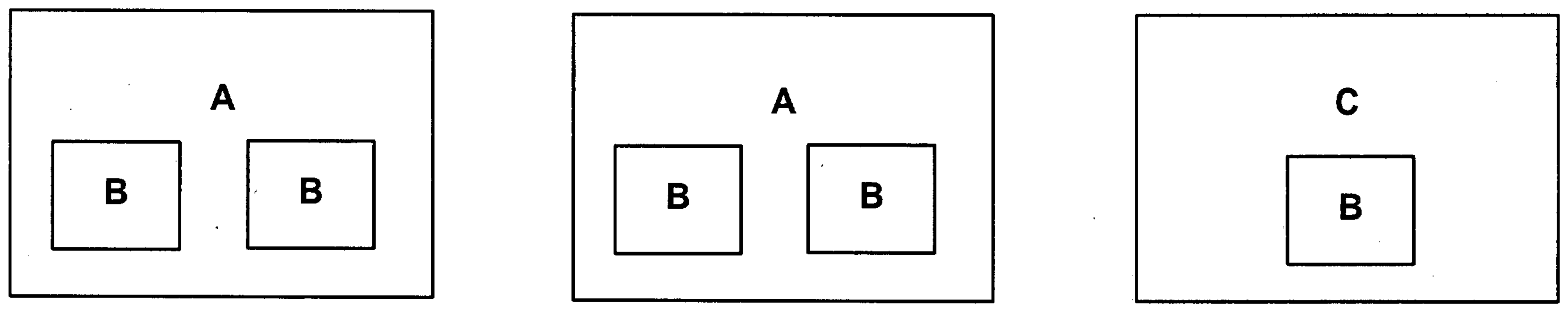

Fig. 7

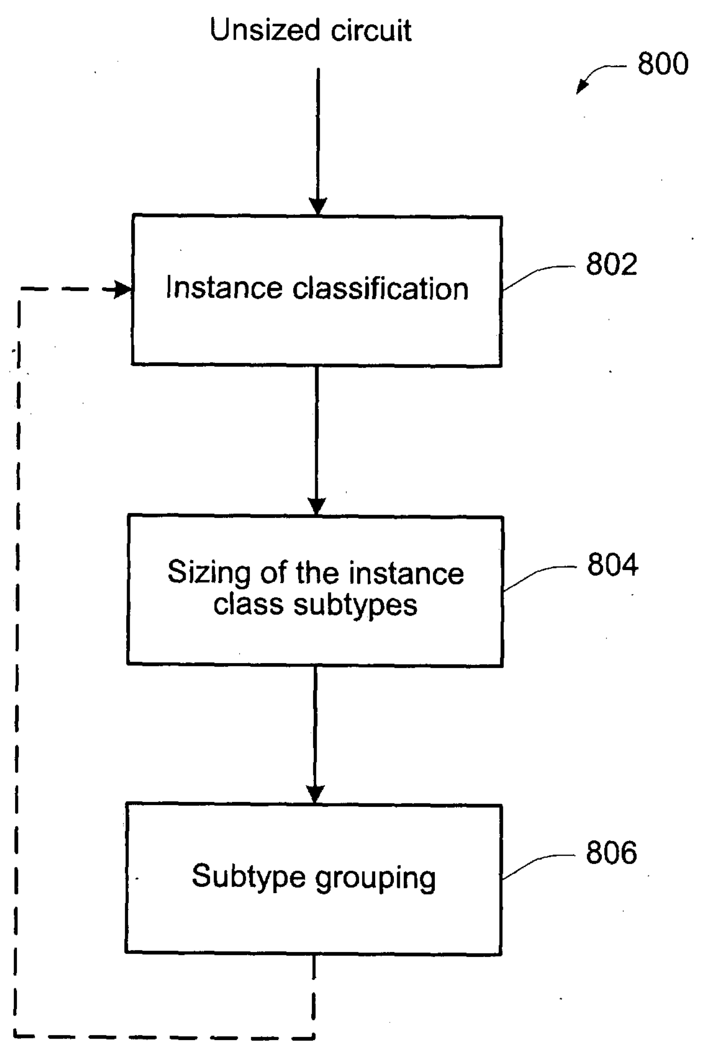
Unsized circuit
800
Instance classification
802
Sizing of the instance class subtypes
804
Subtype grouping
806

Fig. 8

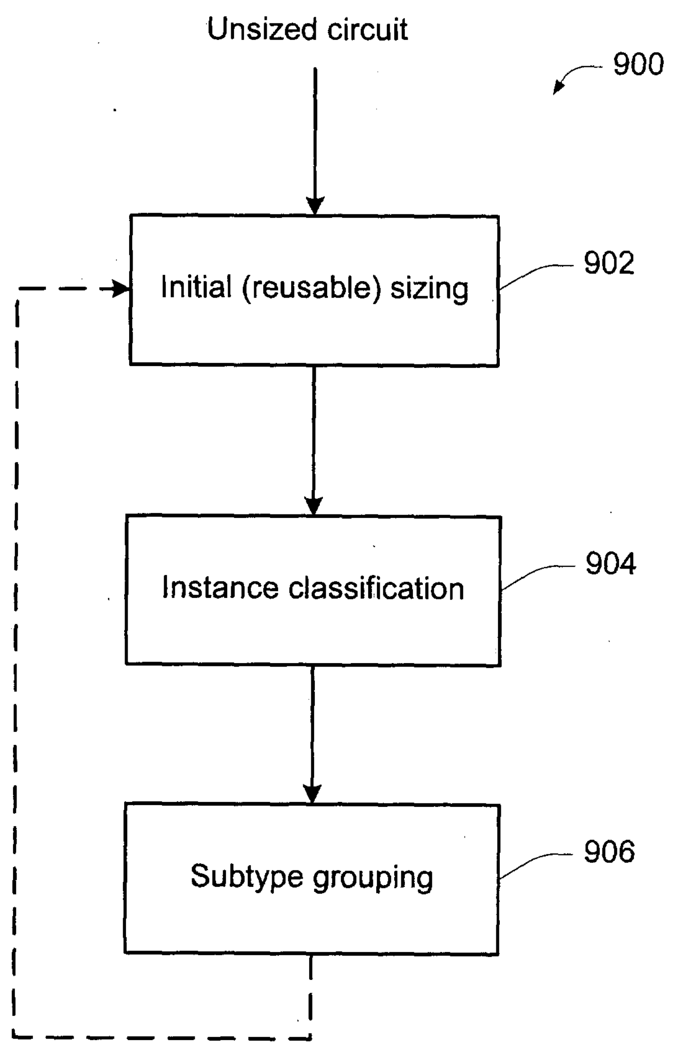
Unsized circuit
900
Initial (reusable) sizing
902
Instance classification
904
Subtype grouping
906

Fig. 9